



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/016,933	12/14/2001	David O. Melgar	RSW920010220US1	6218

7590 12/23/2005
Gerald R. Woods
IBM Corporation T81/503
PO Box 12195
Research Triangle Park, NC 27709

EXAMINER

MITCHELL, JASON D

ART UNIT PAPER NUMBER

2193

DATE MAILED: 12/23/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/016,933	Applicant(s) MELGAR, DAVID O.	
	Examiner Jason Mitchell	Art Unit 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 06 December 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-7, 16, 20, 21, 26 and 27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-7, 16, 20-21, 26-27 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

This action is in response to remarks filed 8/29/05.

At Applicant's request, Claims 1, 16 and 26-27 have been amended. Claims 1-7, 16, 20-21 and 26-27 are pending.

Response to Arguments

Applicant's arguments on pp. 10-12 with respect to claims 1 and 26-27 have been considered but are moot in view of the new ground(s) of rejection.

In the paragraph bridging pp. 10 and 11 Applicant states 'An electronic search of Saulpaugh's text finds only two references to "template"'.

Although Saulpaugh does not use the term "template" his disclosures in col. 24, lines 10-20 ('appropriate reuse of pieces') and lines 37-45 ('a Java dynamic proxy class') each disclose the functionality of a template.

In the first full paragraph on pg. 11, Applicant states:

Furthermore, Applicant finds no teaching in Saulpaugh, nor any suggestion, of "generating code as a class library"

Examiner respectfully disagrees. The use of 'class libraries' was well know in the art, and amounts to little more than the collection of program code to implement various classes. As indicated below in the rejection of claims 1 and 26-27, the teachings of Saulpaugh in col. 23 and 24 would make such a collection obvious to one of ordinary skill in the art.

Art Unit: 2193

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-2, 4-5, 16, 20-21 and 26-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6,792,466 to Saulpaugh et al. (Saulpaugh).

Regarding Claims 1, 26-27: Saulpaugh discloses detecting during run-time processing of a machine-processable definition of a network invocable service, a reference to a structured language specification (col. 20, lines 14-17 'a service advertisement'); locating, responsive to the detection, the referenced structured language specification (col. 20, lines 14-17 'construction of a gate ... from a service advertisement'), the structured language specification encoded in a structured markup language and specifying message syntax definitions for one or more messages usable for interacting with the network-invocable service (col. 20, lines 17-20 'XML service descriptions'); locating, responsive to the detection, a language-specific template that specifies an image for generating code for a particular coding language and specifies where corresponding portions of message syntax definitions are to be substituted therein (col. 24, lines 39-45 'create a Java dynamic proxy class for the service'); and generating the code, according to the template and the definitions in the structured language specification (col. 20, lines 17-20 'a gate factory ... for generating gates based on XML service descriptions'); to be dynamically available for sending request messages to and receiving response messages from, the network-invocable service (col. 18, lines 23-25

'A message gate ... sends and receives type-safe XML messages.),' further comprising steps of: locating, in the structured language specification, the message syntax definitions of the messages; and applying the template to the located message syntax definitions to generate code that, when executed, will build an instance of the message for sending (col. 18, lines 25-28 'Messages gates allow clients and services to exchange XML messages') and will, if the message syntax definition for the message specifies parameters, dynamically obtain values for the parameters and set those parameter values in the built instance (col. 30, lines 10-14 'each method ... containing the marshaled method parameters'); applying the template to the located message syntax definitions to generate code that when executed, will send the built instance of the message, including any set parameter values, to the network-invokable service as a request message (col. 18, lines 25-28 'Messages gates allow clients and services to exchange XML messages'); applying the template to the located message syntax definitions to generate code that, when executed, will receive a response to the sent instance of the message from the network-invokable service as a response message and build a response instance therefrom (col. 18, lines 25-28 'Messages gates allow clients and services to exchange XML messages'); and applying the template to the located message syntax definitions to generate code that, when executed, will dynamically obtain any defined response values from the received response message and populate the response instance therewith (col. 30, lines 10-14 'each method ... containing the marshaled method parameters'); such that the dynamically-generated code is dynamically invocable during the run-time processing for sending the request

Art Unit: 2193

message to and receiving the response message from the network-invokable service.

(Col. 18, lines 25-28 'Messages gates allow clients and services to exchange XML messages').

Saulpaugh does not explicitly disclose the creation of a 'class library'. However, in various embodiments, Saulpaugh does teach 'a cache 141 of gates to avoid constructing them each time the same service is run' (col. 23, lines 65-67) and that 'The generation of gate code at runtime may not be desirable due to memory consumption and code generation time' (col. 24, 46-49). Because cache memory is limited (col. 24, lines 3-6 'the gate cache becomes full') one of ordinary skill in the art would have been motivated to implement the first embodiment (col. 23, line 65-col. 24, line 9) by storing the generated code (col. 24, lines 64-67 'the generation tool is run with input from all the relevant XML schemas for which gates are desired') thereby creating a code library to replace the 'gate cache' freeing up cache memory (col. 24, lines 3-6 'the gate cache becomes full') while avoiding the inflexibility of the second embodiment (col. 24, lines 46-49 'special-purpose clients or small embedded devices').

Further, while Saulpaugh does not explicitly disclose that the 'Java dynamic proxy class' is created with respect to the definitions in the structured language specification, it would have been obvious to a person of ordinary skill in the art at the time of the invention that in order to replace the functionality of the 'gate factory' (col. 24, 39-45 'instead of running the gate factory') the methods generated in this embodiment must also be 'based on XML service descriptions' (col. 20, lines 17-20).

Regarding Claim 2: The rejection of claim 1 is incorporated; further, Saulpaugh discloses the structured language specification is a schema (col. 16, lines 6-7 'A service's message set may be defined using an XML schema').

Regarding Claim 4: The rejection of claim 1 is incorporated; further, Saulpaugh discloses that the structured markup language is Extensible Markup Language (col. 16, lines 6-7 'an XML schema').

Regarding Claim 5: The rejection of claim 1 is incorporated; further, Saulpaugh discloses the message syntax definitions specify elements corresponding to the messages and optionally specify attributes corresponding to the elements, the elements and attributes being encoded in the structured markup language (col. 17, line 66-col. 18, line1 'the messages may include tags ... a message data field').

Regarding Claim 16: The rejection of claim 1 is incorporated; further, Saulpaugh discloses programmatically consulting one or more rules, wherein the rules specify a name for a class library comprising the generated code, to influence processing of the generating step (col. 25, lines 50-53 'gate names may be generated as a combination of a string ... and a random number').

Regarding Claim 20: The rejection of claim 1 is incorporated; further, Saulpaugh discloses the network-invokable service is a web service (col. 15, lines 18-19 'The network may be ... the Internet').

Regarding Claim 21: The rejection of claim 20 is incorporated; further Saulpaugh discloses the reference is specified as a Uniform Resource Locator and the machine-processable definition is specified in a Web Services Definition Language document

(col. 15, lines 23-25 'The advertisement 132 specifies the service's XML schema and URI address').

Claims 3, 6 and 7 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 6,792,466 to Saulpaugh et al. (Saulpaugh) in view of Extensible Markup Language (XML) 1.0 by W3C (XML 1.0).

Regarding Claim 3: The rejection of claim 1 is incorporated; further Saulpaugh does not explicitly disclose the structured language specification is a DTD but discloses that 'A service's message set may be defined using an XML schema' (col. 16, lines 6-7).

XML 1.0 teaches that XML documents are defined by DTDs (2.8 Prologue and Document Type Declaration 'The XML document type declaration ... provide a grammar for a class of documents').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to use such a DTD to define Saulpaugh's messages (col. 16, lines 22-24 'embodied as XML messages') in order to 'provide a grammar' for the messages (XML 1.0 2.8).

Regarding Claim 6: The rejection of claim 5 is incorporated; further, Saulpaugh does not explicitly disclose the message syntax definitions specify at least one child element for at least one element. However Saulpaugh does disclose that 'A service's message set may be defined using an XML schema' (col. 16, lines 6-7) and that the messages are in XML (col. 16, lines 22-24 'embodied as XML messages').

XML 1.0 teaches that XML supports the parent child relationship (2.1 Well-formed XML Documents 'P is referred to as the parent of C, and C as a child of P').

It would have been obvious to a person of ordinary skill in the art at the time of the invention define child elements in Saulpaugh's messages because one of ordinary skill in the art would have been motivated to leverage the XML's full functionality thereby creating a more robust messaging system (col. 14, lines 38-43 'XML may be leveraged').

Regarding Claim 7: The rejection of claim 5 is incorporated; further, Saulpaugh does not explicitly disclose the message syntax definitions specify whether the attributes are required attributes. However Saulpaugh does disclose verifying messages (Col. 7, lines 48-50 'verify the correctness of the message'). Saulpaugh further disclose those messages are in XML format (col. 16, lines 22-24 'embodied as XML messages'). XML 1.0 teaches that XML supports required attributes (3.3.2 Attribute Defaults 'An attribute declaration provides information on whether the attribute's presence is required').

It would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize XML's required attributes because one of ordinary skill in the art would have been motivated to leverage the XML's full functionality thereby creating a more robust messaging system (col. 14, lines 38-43 'XML may be leveraged').

Conclusion

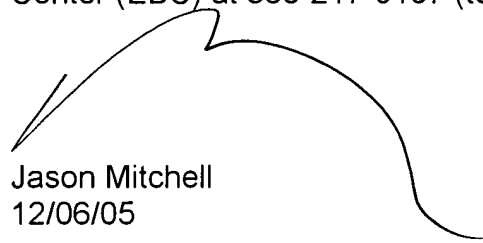
The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Art Unit: 2193


Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571) 272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571) 272-3719. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Jason Mitchell
12/06/05



JOHN CHAVIS
PATENT EXAMINER
ART UNIT 2193